TABLE NO □□□

STUDENT ID NO □□□□□□□□□□

MULTIMEDIA UNIVERSITY

SUBJECT CODE ____________________

MULTIMEDIA UNIVERSITY

FINAL EXAMINATION

TRIMESTER 3, 2017/2018

TSE3351 – SOFTWARE EVOLUTION AND MAINTENANCE

(All sections / Groups)

4 JUNE 2018
9:00 am – 11:00 am
(2 Hours)

Question	Mark
A	
B	
C	
D	
Total	

Examiner 1 Signature: ____________________

Examiner 2 Signature: ____________________

Examiner 3 Signature: ____________________

INSTRUCTIONS TO STUDENTS

1. This question paper consists of 10 printed pages (including cover page) with 4 Sections only.

2. Attempt **ALL** questions in **SECTION A, SECTION B, SECTION C and SECTION D**. The distribution of the marks for each question is given.

3. Please write all your answers **CLEARLY** in the specific answer box provided for each question. Submit this question paper at the end of the examination.

Attempt ALL questions in SECTION A, B, C and D.

Section A (12.5 marks)

Consider a loosely coupled architecture for e-coaching systems (LAES), which addresses the main design concerns of these systems and decouples the system structure from its behaviour. This strategy allows for addressing the design of these systems in an incremental way to obtain a flexible solution that can self-adapt its behavior during the coaching process. The proposed architecture also provides a guide for conceiving e-coaching systems and exploring the particular design concerns (persuasion, self-adaptation, personalization, reasoning, and diagnosing). This represents a first step toward a personal informatics theory that organizes, understands, and contextualizes the knowledge in this study domain, helping to address the study and modeling of these systems in a more affordable way.

Based on the above scenario, answer the following questions Q-A1 to Q-A5:

A1. How does a process of software evolution work for LAES?

(1 mark)

A2. What does software maintenance concerned with? Explain TWO reasons why is software maintenance important for the abovementioned LAES.

(1.5 + 2 marks)

Continued...

A3. If a system such as LAES is used, it is never finished because it will always need to evolve. Why?

(1 mark)

A4. What are the THREE components that define software like LAES as proposed by McDermid? For each component, provide at least TWO examples.

(3 marks)

A5. Explain FOUR factors (with at least an example for each factor) that provide the motivation for software maintenance in LAES.

(4 marks)

Continued…

Section B (12.5 marks)

At NASA's laboratory, the Mission Design and Navigation Software Group (MDN) has two critical systems in operation—a legacy navigation system and its replacement, the Monte system. These systems are in continual operation for most of NASA's deep-space missions. The development and maintenance of them is unquestionably critical to the success of those missions.

With over 800,000 lines of continually evolving code, NASA MDN does not know exactly what or when maintenance issues will surface. MDN must plan to resolve them in a timely manner. The reality is that MDN is continually faced with exceptionally tight resources and schedule constraints. It is impractical to have a standing army of qualified maintenance staff at the ready to attack bugs and implement system enhancements and adjustments on demand. With these constraints and the inherent uncertainty in the demand for maintenance, it is unimaginable how MDN could have successfully sustained Monte over the past 12 years without reliance on their data models and analytics.

B1. Explain TWO reasons why configuration management in software development differs from configuration management in software maintenance for the abovementioned Monte system.

(4 marks)

B2. What are the THREE main responsibilities of the management in Monte change control?

(1.5 marks)

Continued...

B3a. What comprises a good software maintenance framework for the Monte system? What does SPE classification stand for?

(0.5 + 1.5 marks)

B3b. Explain why the SPE classification scheme is useful in software evolution. What are the TWO qualities involving this SPE classification scheme?

(1 + 2 marks)

B4. Explain TWO factors that prevent NASA MDN to undertake domain analysis in using a novel approach such as Microsoft Visual Studio 2017 platform.

(2 marks)

Continued...

Section C (12.5 marks)

C1a. Describe the FOUR stages representation of the spiral model of the development process.

(2 marks)

C1b. Draw the diagram to show THREE spiral cycles, indicating the production process. On each of the THREE spiral cycle line, label the FOUR stages to represent as quadrants for Cycle 1-a, 1-b, 1-c, 1-d; Cycle 2-a, 2-b, 2-c, 2-d; and Cycle 3-a, 3-b, 3-c, and 3-d.

(3 marks)

Continued...

C2. To achieve the objectives of maintenance, a wide spectrum of change to the software product may be necessary. Explain why it is important to categorize software changes.

(2 marks)

C3. There are four types of software change. Define each of the following software change.

C3a. Corrective change.

(0.5 marks)

C3b. Adaptive change.

(0.5 marks)

C3c. Perfective change.

(0.5 marks)

Continued…

C3d. Preventive change.

(1 mark)

C4. Draw a diagram to show the potential relationships between the different types of software changes as mentioned in Question C3 above.

(3 marks)

Continued...

Section D (12.5 marks)

Consider a Wise Development Suite (WDS), where human–machine interaction in system engineering relies mostly on a one-way initiative: the humans instruct or query the computer, while the computer makes an effort to understand and do what the humans had in mind. Moreover, the competencies of the computer are presently more limited than those applied by humans when planning new systems or discussing development tasks. In the future, the WDS will initiate actions and suggestions based on deep knowledge and understanding of a broader range of goals and constraints than is possible today; covering knowledge that is not captured in the specification or code of the system at hand but, rather, comes from domain expertise or general world knowledge and human experience.

The creation of a powerful initial WDS is based on three main cornerstones:

- collecting, representing, and structuring knowledge about systems and problem domains through a common formalism (CF);
- conducting human-computer discourse about systems in natural and appealing ways through a special interaction language (IL); and
- rigorously analyzing and drawing conclusions from these knowledge items and interactions using a dedicated analysis engine (AE).

Based on the above scenario, answer the following questions Q-D1 to Q-D4:

D1. Explain FIVE guidelines in producing good software tools documentation for WDS.

(5 marks)

Continued...

D2. Software measurement is a process to quantify an attribute of a software tool and process. List at least FIVE examples that we can measure the WDS software tool.

(2.5 marks)

D3. Explain FOUR difficulties that you would expect in the context of WDS software reuse.

(2 marks)

D4. Provide SIX benefits that can be derived from WDS software reuse.

(3 marks)

End of Paper